

Article

A Fast Algorithm for Onboard Progressive Flooding Simulation

Luca Braidotti ^{1,2,*}  and Francesco Mauro ^{1,3} 

¹ Department of Engineering and Architecture, University of Trieste, Via Valerio 10, 34127 Trieste, Italy; fmauro@units.it

² Faculty of Engineering, University of Rijeka, Vukovarska 58, 51000 Rijeka, Croatia

³ Department of Naval Architecture, Ocean and Marine Engineering, Maritime Safety Research Centre, University of Strathclyde, Glasgow G4 0LZ, UK

* Correspondence: lbraidotti@units.it

Received: 31 March 2020; Accepted: 20 May 2020; Published: 22 May 2020



Abstract: The need for decision support after a flooding casualty requires the development of fast and accurate progressive flooding simulation procedures. Here, a new quasi-static technique is presented, proposing a differential algebraic formulation capable to consider independently the flooding process in the internal rooms. The proposed method is efficient while simulating long flooding chains along rooms connected by similar size openings, a condition that likely occurs on large passenger ships. Moreover, the computational performances of the simulation procedure have been enhanced by adapting the time step to the progressive flooding pace. The adoption of an adaptive time step algorithm reduces significantly the calculation time. The novel procedure has been tested on the recommended benchmark cases for flooding simulations, highlighting the accuracy and flexibility of the proposed method.

Keywords: progressive flooding simulation; quasi static method; onboard DSS; differential algebraic equations

1. Introduction

In the last decade, the need to provide onboard decision support during a flooding casualty has been highlighted by several accidents. In particular, the sink of Costa Concordia highlighted the vulnerability of large passenger ships and the difficulties associated with the prediction of flooding dynamics after an accident [1]. Large passenger ships have a very complex non-watertight subdivision within the main compartments as well as above the bulkhead deck [2]. The internal subdivision may reduce the free surface moment but lead to possible asymmetric flooding scenarios and to a difficult flooding sequences prediction [3]. Then, after a collision or grounding the crew cannot be sure about the ship final fate (new equilibrium position, capsizing, foundering) and neither of the time-to-flood, i.e., the time to reach the final stage of flooding. Therefore, it is really handy for a master to evaluate the ship dynamics in case of flooding [4]. To this end, use can be made of enhanced onboard Decision Support Systems (DSS) that, after the identification of damage location and extent, are capable to simulate with fast simplified algorithms the flooding dynamics [5,6].

To evaluate the final flooding stage, the more a simulation technique adopts complex formulations devoted to enhancing fidelity to the real event the more the associated computational time increases. The fully-dynamic methods and Computational Fluid Dynamics (CFD) ones require a too high computational time. As an example, one of the fastest dynamic methods is capable to reproduce a model-scale experiment almost three times faster than real-time [7]. These performances, although being remarkable for a dynamic method, are still not acceptable for decision support purposes on a

large passenger vessel, since the evacuation, if required, could be triggered too late. Hence, only a few methods among the ones available in literature can be currently applied onboard for a fast survey, all based on quasi-static approximation [8].

These methods reduce the computational effort by assuming a flat free surface and discarding all the dynamic phenomena other than the floodwater spreading inside the ship [9,10]. Hence, although they cannot reproduce the rough sea and water jet effects [11], they are capable to accurately predict the flooding sequences as well as the time frame of the progressive flooding process [12]. In addition, simplifications can be introduced to further speed up the simulation process. For instance, a simulation 200 times faster than the time-to-flood has been obtained [13]. However, with these assumptions, the reliability of the simulation results can be compromised.

A viable solution to reduce the calculation time while avoiding too simplistic approximations is given by the linearisation process of the governing equations [14]. This method can be further improved by removing some assumptions necessary to solve the progressive flooding dynamic in complex internal geometries. Here, it is proposed to model the dynamic by means of Differential-Algebraic Equations (DAE) system capable to consider independently the flooding state of each internal room. The proposed methodology is then more flexible and accurate compared to the previous process based on an Ordinary Differential Equations (ODE) system. These advantages are highlighted by means of a simple test case. Moreover, the introduction of an adaptive time step for integration enhances the performances in terms of computational effort. The calibration of the adaptive time step algorithm is based on the recommended benchmark test case for progressive flooding simulation validation [15].

2. Basic Simulation Method

The present study is based on a linearised method for progressive flooding simulation [14]. In the present section, the main assumptions of the progressive flooding simulation technique are introduced together with the overall process and the adopted system of governing equations.

2.1. Assumptions and Reference Systems

The proposed simulation process is quasi-static. This means that all the dynamic phenomena other than floodwater pouring are neglected. Thus, the sea free surface, as well as the waterplanes inside flooding compartments, are assumed to be flat and the waterplanes are always parallel to the sea free surface. With these assumptions two reference systems can be defined (Figure 1). The first is the standard ship-fixed right-handed reference system $OXYZ$ having origin at the intersection between baseline and aft perpendicular. The second is an auxiliary earth-fixed right-handed reference system $O'xyz$ corresponding to the sea free surface. In such a system, the flooded compartments' waterplanes can be identified by a level z . This auxiliary system can be defined at each time step according to the ship floating position, i.e., the transitory equilibrium position evaluated taking into account the initial loading condition and the embarked floodwater.

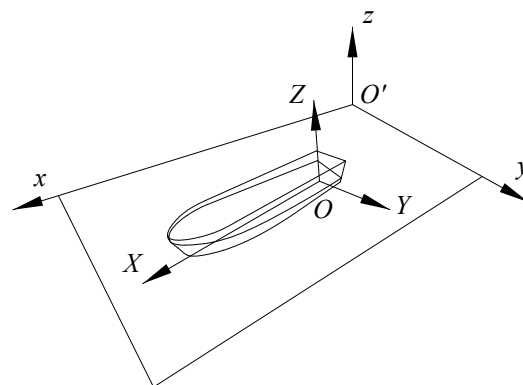


Figure 1. Ship-fixed and auxiliary earth-fixed reference systems [14].

All the internal spaces are assumed to be fully-vented. Air compression is significant only in rooms connected by small size vent openings, e.g., tanks equipped with ventilation pipes [16]. Usually, ship 3D models adopted for stability assessment do not include such details, in order to streamline the flooding model configuration. In the main compartments, stable air pockets can appear when all the openings are flooded. However, the internal watertight boundaries are not always completely airtight [17], leading to a difficult assessment of effective vent area. Finally, in a real environment, ship model uncertainties (e.g., permeabilities, areas of the openings, their constant discharge coefficients and leakage/collapse water head, etc.) might anyway affect simulation reliability [18,19], vanishing the accuracy enhancement coming from air compression modelling.

2.2. Main Calculation Loop

The main iterative loop of the simulation process, sketched in Figure 2, is composed by the following steps. First, the ship displacement and centre of mass are assessed in order to evaluate the current floating position [20]. The floating position is considered fixed over a single time step allowing to define the temporary earth-fixed reference system. When no feasible equilibrium is found, the ship sinks or capsizes stopping the simulation process. Otherwise, the new water levels z^* inside the flooded rooms are evaluated by means of a 1-dof equilibrium procedure in order to assure the conservation of the floodwater volume, as assessed in the previous step, inside each room.

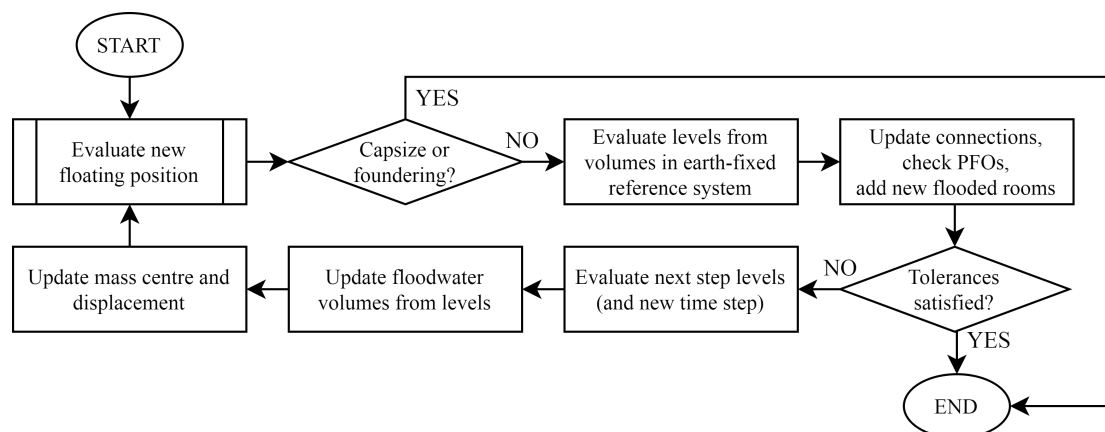


Figure 2. Flowchart of the main loop of the simulation process.

According to these initial water levels, all the openings connecting a flooded room to an empty one (hereinafter referred as Progressive Flooding Openings PFO) are checked. If any PFO is submerged, a new flooded room has to be considered in the simulation. When dealing with very complex internal layout resulting in asymmetric flooding, counter flows could arise too. In such cases, a previously flooded room can be again emptied or a previously submerged opening can emerge. Therefore, the status of the openings connecting two flooded rooms (hereinafter referred as connections) has to be checked, moving the emerged connections to the PFOs list and removing the empty rooms. In special cases, some floodwater can be trapped in a room by the surfacing of all the connections. The room has still to be removed from the flooded rooms and treated as a tank in the next equilibrium assessment.

If no new flooded rooms are detected, equilibrium is checked to detect when the ship has reached a new stable floating position. To this end, two stopping conditions have to be satisfied. The first is related to the rate of change of the heel angle ϕ , the trim angle θ and the mean draught T_M . The second is related to the level inside the non-full flooded rooms. According to the definition of the auxiliary reference system, a null floodwater level corresponds to the flat sea surface one. Hence, when all the levels are null (with a small tolerance) the second condition is satisfied. Table 1 provides the adopted tolerances for the stopping criteria.

Table 1. Applied thresholds of the stopping algorithm.

Description	Bounded Value	Threshold
Heel Angle	$ \phi_n - \phi_{n-1} /dt$	0.00050 deg/s
Trim Angle	$ \theta_n - \theta_{n-1} /dt$	0.00005 deg/s
Mean Draught	$ T_{M_n} - T_{M_{n-1}} /(T_{M_0} dt)$	0.00001 1/s
Level	$ z_i /T_{M_0}$	0.0001

If convergence conditions are not satisfied, the next time step levels have to be assessed. The new levels can be evaluated with different integration procedures. Eventually, the next-step water volumes inside flooded rooms can be evaluated from the corresponding levels, allowing to start a new main loop iteration.

2.3. ODE System

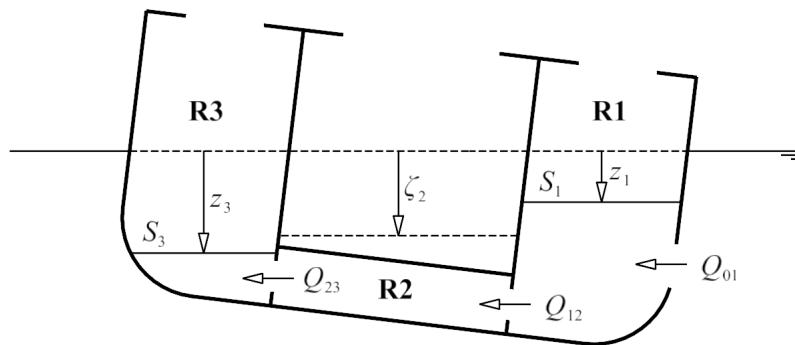
The progressive flooding dynamic is governed by two main equations: the conservation of mass and the conservation of momentum, that have to be instantaneously satisfied for each flooded room and for each connection among them. Considering the i -th room, the conservation of mass states that the sum of incoming and outgoing fluxes is equal to the volume derivative:

$$\dot{V}_i = \sum_{j=1}^{O_i} Q_{ji} \quad (1)$$

where \dot{V}_i is the time derivative of floodwater volume inside the room and Q_{ji} are the flow rates through the O_i openings connecting the room to other ones, denoted by index j . Assuming a constant waterplane area, the conservation of mass can be written as:

$$\dot{z}_i \mu_i S_i = \sum_{j=1}^{O_i} Q_{ji} \quad (2)$$

where S_i , μ_i and \dot{z}_i are the waterplane surface of the i -th room, its permeability and its floodwater level derivative, respectively (Figure 3), all referred to the auxiliary reference system. It is worth to notice that this assumption is not a threat for slow heeling variations within wall-sided spaces that are extremely common in the subdivision layout of large passenger ships.

**Figure 3.** Sketch of a three rooms geometry.

The conservation of momentum describes the floodwater flowing through any connection. Assuming a stationary flow, the Bernoulli equation can be adopted for a streamline starting from the centre of the room's free surface A and ending to the opening centre B :

$$\frac{p_B - p_A}{\rho} + \frac{1}{2} (u_B^2 - u_A^2) + g(z_B - z_A) + \frac{1}{2} k_L u_B^2 = 0 \quad (3)$$

where p is the air pressure, u the flow velocity and k_L the opening's pressure loss coefficient. Neglecting air compression and assuming a null flow velocity in free surface centre $u_A = 0$, the flow rate through the opening can be evaluated as:

$$Q_{ji} = C_{dji} A_{ji} \operatorname{sgn}(z_j - z_i) \sqrt{2g|z_j - z_i|} \quad (4)$$

where A_{ji} are the opening areas and $C_{dji} = 1/\sqrt{1+k_L}$ are non-dimensional discharge coefficients depending on the orifice shape and edges.

Considering a generic initial time instant t^* and n partially filled rooms, combining the Equations (2) and (4), a system of ODE can be defined in the form:

$$\dot{\mathbf{z}}(t - t^*) = \tilde{\mathbf{f}}(\mathbf{z}(t - t^*)) \quad (5)$$

where $\tilde{\mathbf{f}}$ is:

$$\tilde{f}_i(\mathbf{z}(t - t^*)) = \sum_{\substack{j=0 \\ j \neq i}}^n \frac{K_{ji}}{\mu_i S_i} \operatorname{sgn}(z_j - z_i) \sqrt{|z_j - z_i|} \quad (6)$$

and

$$K_{ji} = C_{dji} A_{ji} \sqrt{2g} \quad (7)$$

The system can be solved with different strategies, obtaining the new floodwater levels to be used in the new main-loop iteration.

The ODE formulation gave good results in simulating the progressive flooding of partially filled rooms. However, if some of them are completely filled, the volume of floodwater inside them should be kept constant in order to satisfy the conservation of mass. Means that $S_i \dot{z}_i = 0$ and thus the system is no more ordinary. To resolve it as an ODE system, a non-null free surface area has to be assumed. For instance, it can be considered equal to the room's top area, accepting mass conservation violation. This issue can be overcome forcing the grouping of the completely filled room with the one connected by the larger area opening [14]. However, such an approach neglects the pressure loss of the opening connecting the two grouped rooms leading again to inaccurate simulation results.

3. DAE System

To overcome the ODE formulation restrictions, an alternative strategy is required. In the present section, the accuracy of the linearised simulation technique is enhanced by introducing a DAE system formulation. Its advantages are here highlighted by comparing the results coming from the previous procedures on a simple geometry.

As mentioned, introducing $m - n$ completely filled rooms, the volume of floodwater inside them should be kept constant in order to satisfy the conservation of mass. Therefore, the related left terms in the System of Equation (5) are null. This means that the ODE system becomes a system of non-linear semi-explicit differential-algebraic equations:

$$\begin{cases} \dot{\mathbf{z}}(t - t^*) &= \mathbf{f}(\mathbf{z}(t - t^*), \boldsymbol{\zeta}(t - t^*)) \\ \mathbf{0} &= \mathbf{g}(\mathbf{z}(t - t^*), \boldsymbol{\zeta}(t - t^*)) \end{cases} \quad (8)$$

where the \mathbf{z} are still the floodwater levels of inside the n partially filled rooms while $\boldsymbol{\zeta}$ are the water heads inside the $m - n$ completely filled ones, both expressed in the earth-fixed reference system. The functions \mathbf{f} and \mathbf{g} can be written as:

$$f_i(\mathbf{z}(t-t^*), \boldsymbol{\zeta}(t-t^*)) = \sum_{\substack{j=0 \\ j \neq i}}^n \frac{K_{ji}}{\mu_i S_i} \operatorname{sgn}(z_j - z_i) \sqrt{|z_j - z_i|} + \quad (9a)$$

$$+ \sum_{j=n+1}^m \frac{K_{ji}}{\mu_i S_i} \operatorname{sgn}(\zeta_j - z_i) \sqrt{|\zeta_j - z_i|}$$

$$g_i(\mathbf{z}(t-t^*), \boldsymbol{\zeta}(t-t^*)) = \sum_{j=0}^n K_{ji} \operatorname{sgn}(z_j - \zeta_i) \sqrt{|z_j - \zeta_i|} + \quad (9b)$$

$$+ \sum_{\substack{j=n+1 \\ j \neq i}}^m K_{ji} \operatorname{sgn}(\zeta_j - \zeta_i) \sqrt{|\zeta_j - \zeta_i|}$$

3.1. Differential Part

In order to enhance the computational performances, the DAE system can be solved separating the differential and the algebraic parts. The differential part can be linearised to obtain an analytical solution. This approach allows to increase the integration time step with respect to the ones required by standard integration techniques, like Runge–Kutta, ODE-45 or Runge–Kutta–Fehlberg, to solve the non-linear system. Therefore, an overall reduction of the total calculation time for the progressive flooding simulation process is achieved.

The differential part of the System in Equation (8) can be also linearised in $\mathbf{z}^* = \mathbf{z}(t^*)$ as follows:

$$\dot{\mathbf{z}}' = \mathbf{J}_f(\mathbf{z}^*, \boldsymbol{\zeta}^*) \mathbf{z}' + f(\mathbf{z}^*, \boldsymbol{\zeta}^*) \quad (10)$$

where $\mathbf{z}' = \mathbf{z} - \mathbf{z}^*$ and \mathbf{J}_f is the Jacobean matrix of the function f evaluated in $(\mathbf{z}^*, \boldsymbol{\zeta}^*)$. The linearised system can be solved according to [14], obtaining an explicit solution for the water level evolution inside the partially filled rooms as a function of time.

3.2. Algebraic Part

As the time step dt is defined, the algebraic solution provides the next iteration levels $\mathbf{z}(dt)$. Hence, the water head inside the completely filled rooms can be also determined by solving in $\boldsymbol{\zeta}(t)$ the algebraic component of the System in Equation (8). The system is non-linear and differentiable, being its Jacobean matrix \mathbf{J}_g defined as:

$$J_{g_{ij}} = \begin{cases} -\sum_{k=0}^n \frac{K_{ki}}{\sqrt{4|z_k(dt) - \zeta_i|}} - \sum_{\substack{k=n+1 \\ k \neq i}}^m \frac{K_{ki}}{\sqrt{4|\zeta_k - \zeta_i|}} & \text{if } i = j \\ \frac{K_{ji}}{\sqrt{4|\zeta_j - \zeta_i|}} & \text{if } i \neq j \end{cases} \quad (11)$$

Thus, the system can be solved as a least-square problem. In the present study, the Levenberg–Marquardt algorithm has been applied, assuming as initial condition $\boldsymbol{\zeta}^* = \boldsymbol{\zeta}(t^*)$. The water heads which satisfy the minimum condition are assumed as system solution $\boldsymbol{\zeta}(dt)$, completing the unknown variables definition for next main-loop iteration. Hence, with the linearised DAE formulation (LDAE), each completely filled room can be independently modelled, removing the need for grouping required by linearised DOE formulation (LODE).

3.3. Test Case

The proposed method has been applied to a test case in order to show the effect of DAE formulation. Figure 4 shows a sketch of the adopted test geometry including the flooding sequence and in Table 2 the main dimensions are provided. The box-shaped geometry is divided into five rooms

that are assumed to be fully-vented. The tanktop of the compartments is located at 5 m above baseline and 7 m below the external free surface. The position of the test arrangement is considered fixed, therefore no equilibrium assessment is needed. The rooms are interconnected as specified in Table 3.

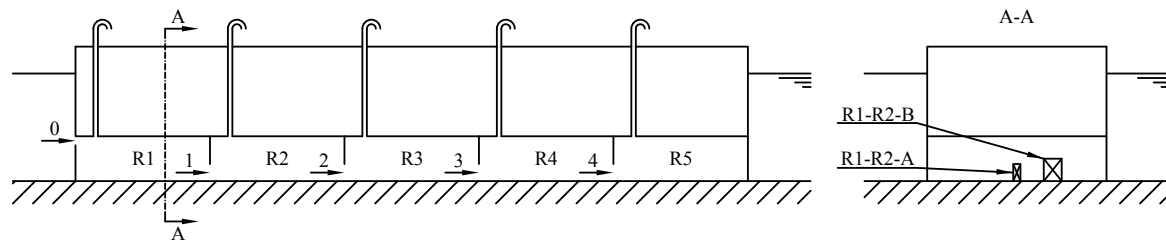


Figure 4. Test case arrangement with flooding sequence.

Table 2. Main particulars of test case arrangement.

Description	Symb.	Value	Description	Symb.	Value
Length overall	L_{OA}	75.000 m	Breadth	B	20.000 m
Draught	T	12.000 m	Height	D	15.000 m

Table 3. Main characteristics of the test case openings. $C = (X_C, Y_C, Z_C)$ is the centre of the opening in ship-fixed reference system.

PFO	h (m)	w (m)	X_C (m)	Y_C (m)	Z_C (m)
SEA-R1	1.000	20.000	0.000	0.000	4.500
R1-R2-A	1.900	0.800	15.000	0.000	4.500
R1-R2-B	2.500	2.000	15.000	0.000	4.500
R2-R3	1.900	0.800	30.000	0.000	4.500
R3-R4	1.900	0.800	45.000	0.000	4.500
R4-R5	1.900	0.800	60.000	0.000	4.500

Two test scenarios have been simulated: the first is a quite long flooding chain of four completely filled rooms connected by equal-size openings (opening R1-R2-B is kept close). Such an arrangement has been selected to reproduce a realistic condition for passenger ships, where the rooms inside a single watertight compartment are usually connected by standard size openings, usually fire doors or other lighter doors having lower flooding restriction. In the second scenario, R1 and R2 are connected by the larger opening R1-R2-B instead of R1-R2-A to study the effect of unequal sized connections. The LDAE results are compared with the LODE ones [14]. They are provided in Figures 5 and 6 for scenarios 1 and 2, respectively, showing the floodwater levels inside the five rooms in the earth-fixed reference system as a function of time.

In the first scenario, the two techniques provide comparable results up to the complete filling of R2. Afterwards, the progressive flooding pace presents relevant differences. In particular, due to the room grouping, the LODE method underestimates the time to flood of about 25% with respect to LDAE one. In fact, applying LODE method, at each room filling the water head is assumed equal to the external free surface, neglecting the pressure loss related to the connection among grouped rooms. Such an assumption might be acceptable only for different sized internal openings, since the pressure loss is proportional to the square of the flow velocity and thus can be negligible for the larger ones. It is the case of the openings SEA-R1 and R1-R2-A, which leads to comparable results in the initial phase. Then, when R2 has been grouped with the external environment, the neglected pressure loss related to R1-R2-A is, in this particular case, equal to the R2-R3 one. Hence, the time pace simulated by the LODE is faster than the LDAE one. The accuracy dependence on the connections' size of the LODE technique is confirmed by the second simulated scenario. Here, the increased size of the R1-R2-B reduces the water head overestimate after R2 filling, extending the comparable results up to R3

filling. Then, again the grouping-related issues can be noted, leading to an overall underestimation of time-to-flood of about 15% applying the ODE formulation. Hence, the LDAE method is more reliable when it is necessary to deal with completely filled rooms compared with the ODE one, which is only capable to provide a qualitative simulation.

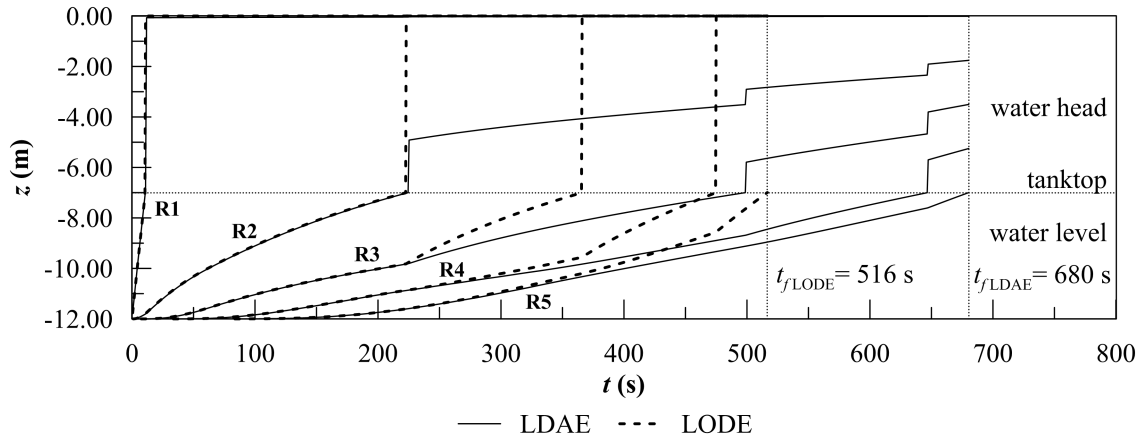


Figure 5. Water levels/heads in earth-fixed reference system on test case arrangement in scenario 1.

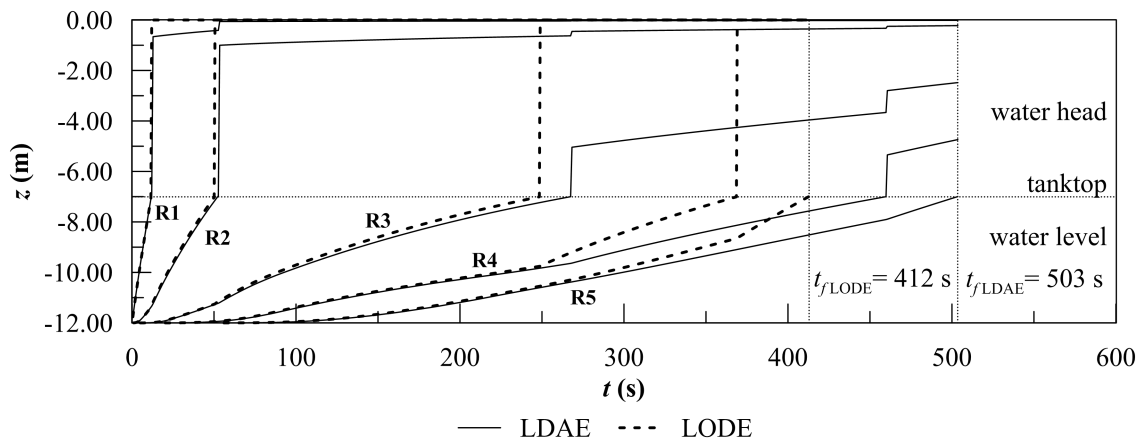


Figure 6. Water levels/heads in earth-fixed reference system on test case arrangement in scenario 2.

Nevertheless, aiming at direct onboard application, it is necessary to study the effects on computation time related to the DAE formulation. Table 4 provides a summary of the mean computation time per iteration related to the primary tasks of the simulation main-loop evaluated for an Intel® Core™ i7-4710 CPU (2.50 GHz) standard laptop. In detail, t_{Vz} is the time to evaluate the levels from volumes, t_{up} the one for model updating, t_{sv} the one required to solve the system, t_{zV} the one to evaluate new volumes from levels and t_{gr} time for grouping. It can be noted that the more relevant contribution is given by the 1-dof equilibrium procedure adopted for the evaluation of the new levels. Considering only the solution time t_{sv} , the LDAE method is slower than the previous one, due to the different dimension of the equations' system. Globally, the LODE method is about 9% faster than LDAE on the two test cases, although the frequent call of grouping algorithm reduces the gain due to system dimension. Thus, the higher accuracy and flexibility of the DAE formulation implies only a moderate drawback on computational load considering a single main-loop iteration.

Table 4. Comparison of mean computational load per loop related to the different simulation techniques.

	t_{Vz} (ms)	t_{up} (ms)	t_{sv} (ms)	t_{zV} (ms)	t_{gr} (ms)	t_{tot} (ms)
LDAE	86.56	0.76	0.53	0.35	0.00	88.21
LODE	75.93	0.78	0.37	0.36	2.25	80.16

4. Computation Time Reduction

The progressive flooding is usually characterised by different stages [21]. In the initial phase, the floodwater rushes into the breached compartments. If the breaches are sufficiently large and deeply submerged, the flooding of the compartment can be considered almost instantaneous. In this phase dynamic phenomena usually arise, being non-negligible and might lead to the ship capsizing [22]. The methods based on quasi-static assumption neglect these effects. Nevertheless, to simulate the fast floodwater level rising in this phase, a small integration time step has to be adopted [23]. In the subsequent phases, the process slows down as the floodwater spreads into other rooms through non-watertight openings. During this transient, the quasi-static hypothesis provides very reliable results and larger time steps could be applied assuring good simulation accuracy [24]. However, the collapse of non-watertight openings, such as fire-doors or splash-doors, as well as the lolling before the ship capsize may drive to sudden process speed-up. Hence, the variable pace of the progressive flooding suggests the adoption of a non-constant integration time step.

4.1. Adaptive Time Step

Here, it is proposed to adjust the time step on the base of the floodwater-levels derivatives related to the partially-filled rooms, that are a good indicator for time step adjustment [24]. In the LDAE method, the initial derivatives \dot{z}^* can be evaluated according to Equation (9a), being determined at each initial time instant t^* . The proposed formulation for the LDAE adaptive time step is:

$$dt = k \frac{T_M^*}{\max \dot{z}^*} \quad (12)$$

where k is a non-dimensional coefficient that can be experimentally calibrated in order to obtain the desired accuracy of the integration process. A metric scaling factor is also necessary to assure the formulation dimensional consistency. Among several tested options (e.g., the main dimensions of the ship and the submersion of the breach) the mean draught at the beginning of integration step T_M^* has been chosen.

4.2. Maximum Time Step Limitation

Although an adaptive time step can reduce the computation time, the application of too large time steps might affect simulation accuracy. The detection of the submersion of a PFO, and thus the addition of a new flooded room, could be delayed, distorting the simulation results. Moreover, the room-filling detection could be also delayed, leading to a late switch from the differential to the analytic formulation. As an example, the Figure 7 shows this drawback on the previous test case (Section 3.3). Finally, an excessive increment of floodwater volumes can lead to a too large change in the ship floating position conflicting with the assumption of fixed floating position over the single time step. All these problems could be mitigated detecting their occurrence a-posteriori and, then, recalculating the last steps with a properly reduced time step. However, aiming to the reduction of computational effort, such an additional iteration should be avoided choosing a methodology that does not imply a recalculation. A viable solution can be defined a-priori a maximum time step dt_{max} based on the results of the latest integration steps.

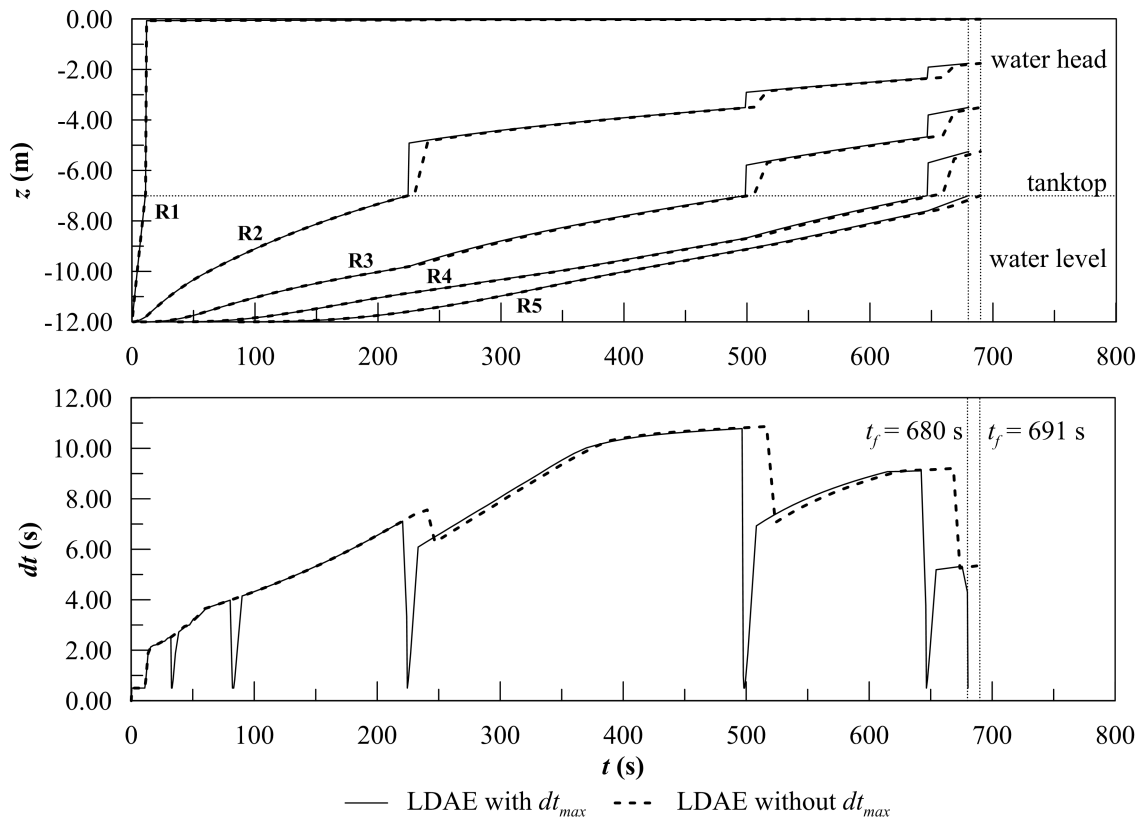


Figure 7. Simulation distortion due to late switch from differential to algebraic formulation on test case arrangement in scenario 1 with adaptive time step ($k = 0.01$).

As an example, considering one PFO connecting a partially flooded room to another, it will be submerged when its lower edge is reached by the level inside the floodwater level. The rising of floodwater level z is described by the solution of the linearised System in Equation (10), which is not invertible. Therefore, to avoid an iterative solution, an estimation of the t_{PFO} can be obtained by exponential fitting of the levels coming from the last two time steps (Figure 8). Under this assumption, the estimated level equation is:

$$\tilde{z}(t) = z^{**} \left(\frac{z^*}{z^{**}} \right)^{\frac{t}{dt^*}} \quad (13)$$

where dt^* and z^* are the last time step and level, respectively and z^{**} is the penultimate level. The Equation (13) is invertible, therefore, the t_{PFO} can be evaluated analytically as well as the maximum time step related to the PFO (defined as $dt_{PFO} = t_{PFO} - dt^*$):

$$dt_{PFO} = \left[\frac{\ln \left(\frac{z_{PFO}}{z^{**}} \right)}{\ln \left(\frac{z^*}{z^{**}} \right)} - 1 \right] dt^* \quad (14)$$

An analogous approach can be applied to the higher point of flooded rooms to predict filling instant as well as to avoid too large changing of the floating position during a single time step. The maximum accepted values for the variation of the heel angle, trim angle and mean draught from values at the previous time step are reported in Table 5.

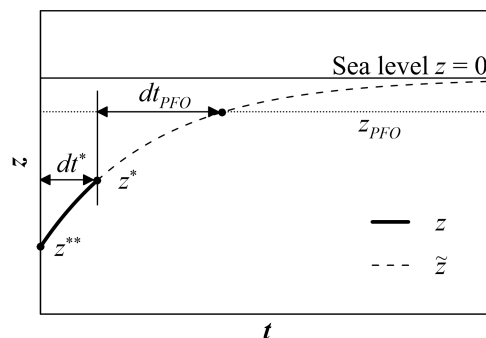


Figure 8. Estimation of time-to-Progressive Flooding Opening (PFO)-submersion for the maximum time step assessment.

Table 5. Maximum accepted variation of floating position per iteration.

Description	Bounded Value	Threshold
Heel Angle	$ \phi_n - \phi_{n-1} $	0.100 deg
Trim Angle	$ \theta_n - \theta_{n-1} $	0.050 deg
Mean Draught	$ T_{M_n} - T_{M_{n-1}} / (T_{M_0})$	0.005

5. Application of the Proposed Methodology

In order to perform the calibration of the non-dimensional coefficient k governing the adaptive time step algorithm, experimental data in model-scale have been adopted [25]. The test arrangement is a box-shaped barge (Figure 9) having the main particulars reported in Table 6. The barge was arranged with two symmetric compartments, split into interconnected rooms by a double bottom and the main deck. In the fore compartment, two symmetric wing tanks have been also fitted on the lower deck. The details about the PFOs are provided in Table 7, including their experimentally determined discharge coefficients. The double bottom in the aft compartment (DB1) was airtight, leading to the appearance of a stable air pocket, having a strong influence on the progressive flooding process. For this reason, a simplified air-pocket modelling approach has been adopted [14].

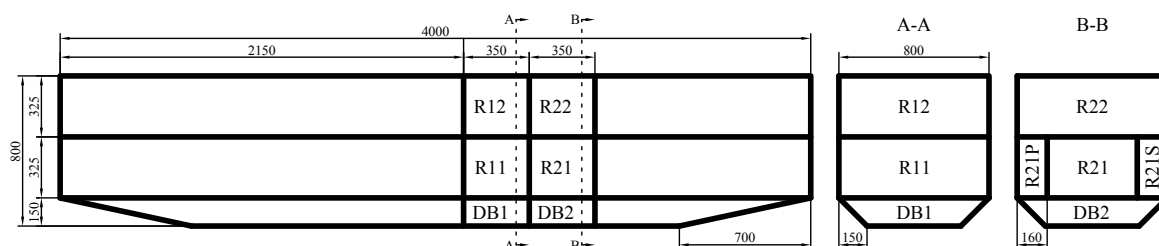


Figure 9. Adopted hull and rooms meshes of the box-shaped barge [14].

Table 6. Main particulars of box-shaped barge.

Description	Symb.	Value	Description	Symb.	Value
Length overall	L_{OA}	4.000 m	Breadth	B	0.800 m
Draught	T	0.500 m	Height	D	0.800 m
Volume	∇	1.450 m ³	Metacentric Height	GM	0.110 m

Table 7. Main characteristics of the barge PFOs. $C = (X_C, Y_C, Z_C)$ is the centre of the PFO in ship-fixed reference system.

PFO	Type	Size (m)	X_C (m)	Y_C (m)	Z_C (m)	C_d (-)
SEA-DB1	Rectangular	0.060×0.040	2.325	0.000	0.005	0.78
SEA-DB2	Rectangular	0.025×0.025	2.675	0.000	0.005	0.83
SEA-R21S	Rectangular	0.060×0.040	2.675	0.395	0.315	0.78
DB1-DB2	Circular	$D = 0.020$	2.500	0.000	0.075	0.80
DB2-R21	Rectangular	0.060×0.040	2.800	-0.170	0.155	0.78
R21-R21S	Rectangular	0.020×0.200	2.675	0.240	0.260	0.75
R21-R21P	Rectangular	0.020×0.200	2.675	-0.240	0.260	0.75
R21-R11	Circular	$D = 0.020$	2.500	0.000	0.315	0.80
R21-R22	Rectangular	0.100×0.100	2.750	-0.160	0.475	0.72
R11-R12	Rectangular	0.100×0.100	2.400	-0.160	0.475	0.72
R12-R22	Rectangular	0.080×0.200	2.500	0.000	0.580	0.72

Three different damage scenarios (A, B, C) have been simulated in the time domain on the box-shaped barge. Figure 10 provides a sketch of the flooding sequences in the three scenarios. More details about the experimental layout, the adopted procedures and a complete description of the studied scenarios can be found in [25].

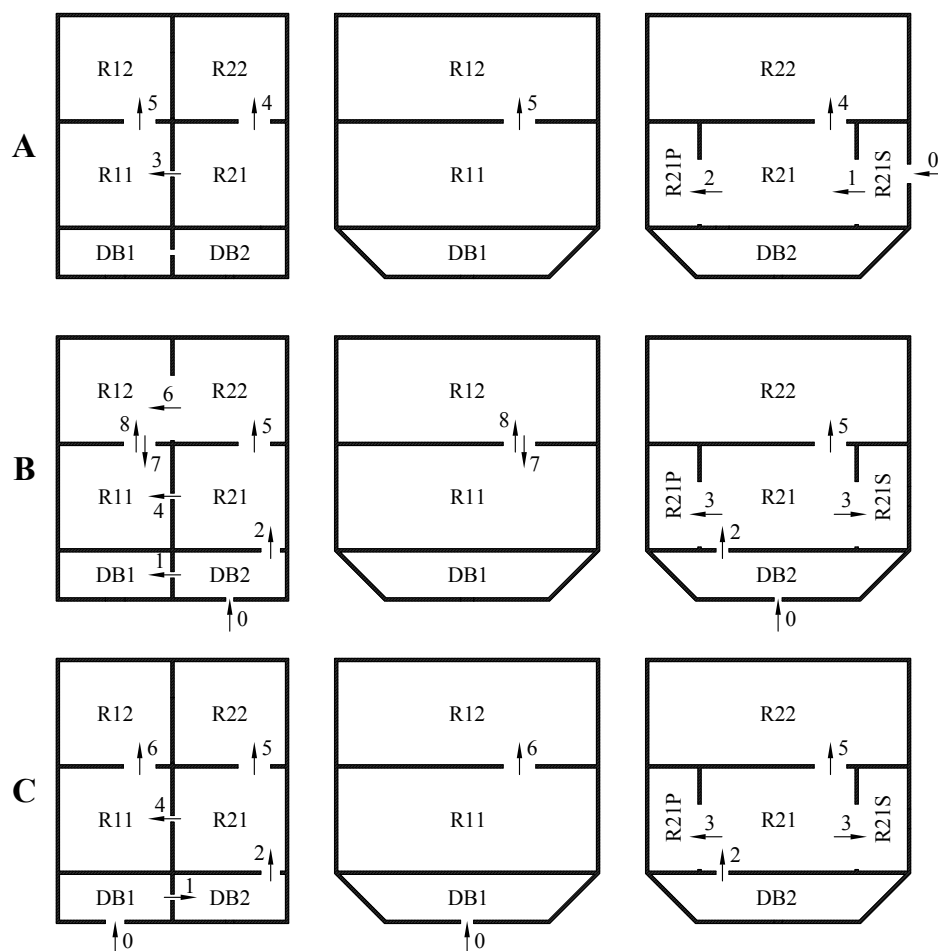


Figure 10. Flooding sequences of the simulated damage scenarios A, B and C.

5.1. Adaptive Time Step Calibration

In order to calibrate the adaptive time step algorithm, simulations of the three damage scenarios were carried out with the following values of non-dimensional coefficient $\mathbf{k} = (0.005, 0.010, 0.025, 0.050)$. This choice is based on preliminary experiences on simple geometries in order to investigate a wide range of time steps. Then, the effect of k on the floating position evolution in time domain has been studied defining an error ratio e_h representing the gap between the experimental records and the simulation:

$$e_{h_i}(t) = \frac{h(t, k_i)}{h_e(t)} \quad (15)$$

where h is the considered dimension at the instant t and h_e the experimental moving average evaluated at t . The ratios e_θ and $e_{\Delta T}$ related to heel angle, trim angle and sinkage, respectively, have been considered. The results of the analysis are provided in Figures 11–13 for damage scenarios A, B and C, respectively. Moreover, in order to investigate the results related to heel angle and the convergence at final stage of flooding a dimensional square error has been defined as follows:

$$e_{h_i}^2(t) = [h(t, k_i) - h_e(t)]^2 \quad (16)$$

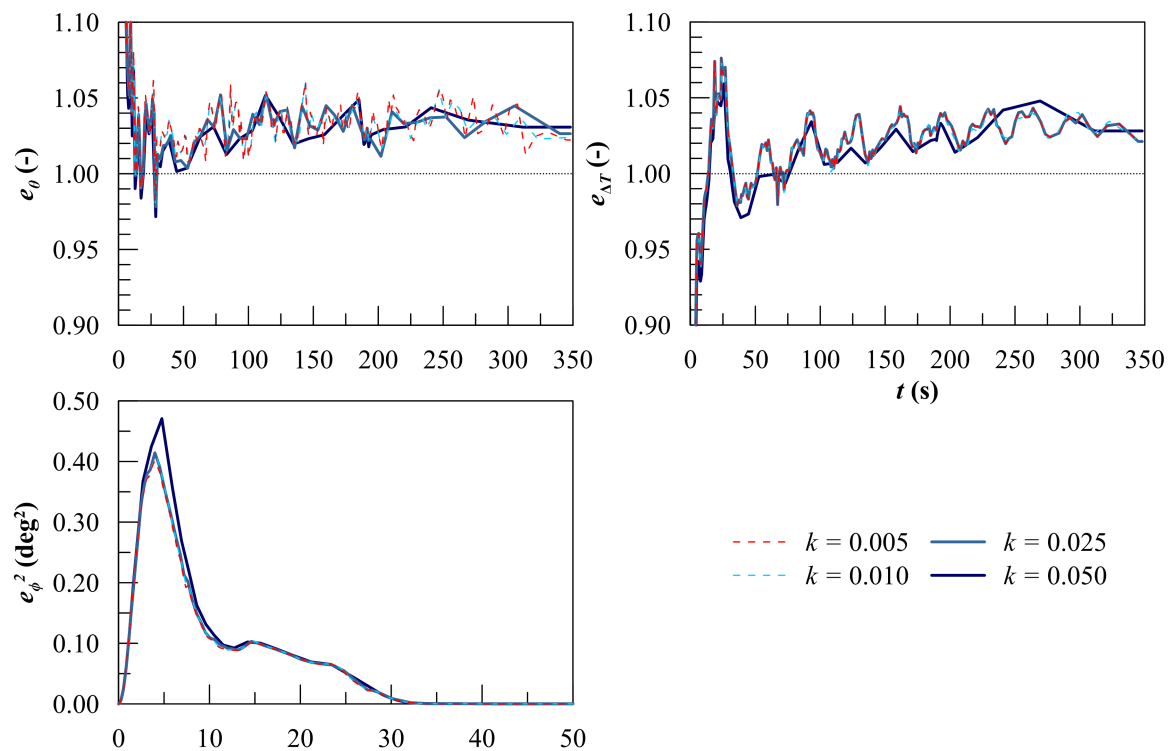


Figure 11. Heel angle squared error, trim angle and sinkage error ratios in scenario A.

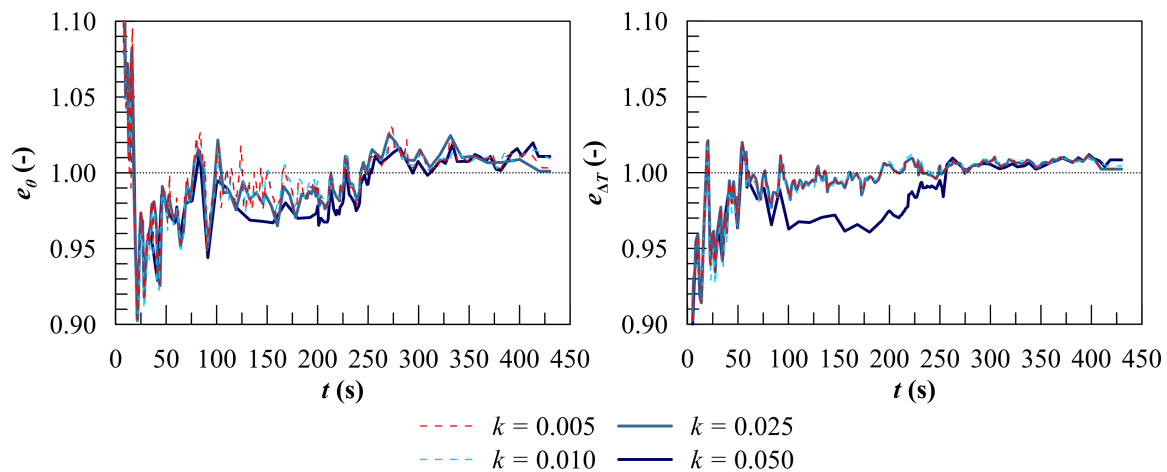


Figure 12. Trim angle and sinkage error ratios in damage scenario B.

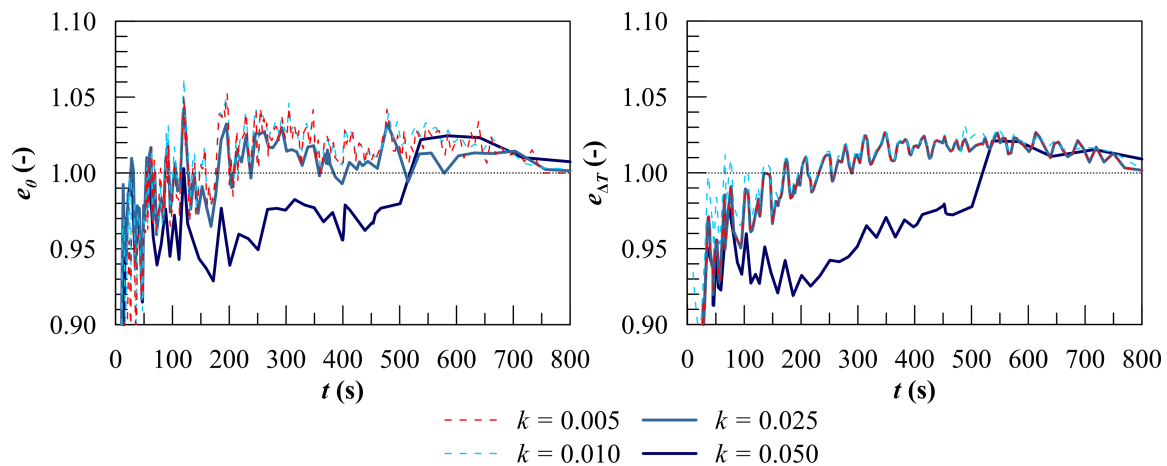


Figure 13. Trim angle and sinkage error ratios in damage scenario C.

In Figure 11 is reported the squared error for heel angle in scenario A. In addition, Figures 14–16 provides the values of the errors at the final equilibrium stage. They are compared with the mean squared error MSE evaluated on the tail of the experimental record in the final equilibrium position:

$$MSE = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n} \quad (17)$$

where \bar{x} is the mean value evaluated on the record tail.

Considering the benchmark test case, it is worth to notice that the squared errors converge to the values coming from the experimental records at final stage of flooding. In detail, in all the scenarios, the squared errors related to $k = 0.005$ and $k = 0.010$ are lower than the MSE evaluated on the experimental record tail. Considering the whole progressive flooding process in the time domain, the error ratios related to the trim angle and the sinkage are always lower than 0.05, excluding the initial phase of the simulations, where the close-to-zero absolute values of trim and sinkage drive to their divergence. Moreover, the error ratios related to $k = 0.005$ and $k = 0.010$ have very similar trends for all the damage scenarios, hence, the value $k = 0.010$ can be considered a good parameter to assure simulation consistency when high accuracy is required.

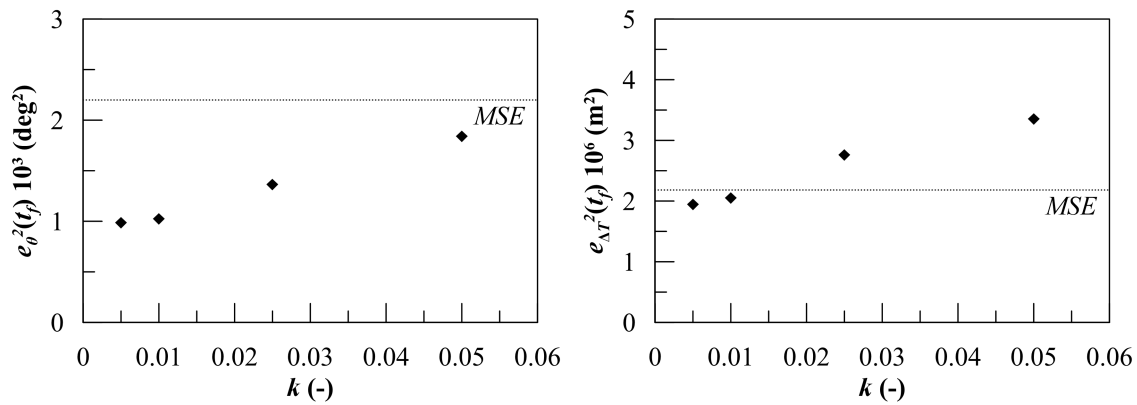


Figure 14. Trim angle and sinkage squared error at final stage in scenario A.

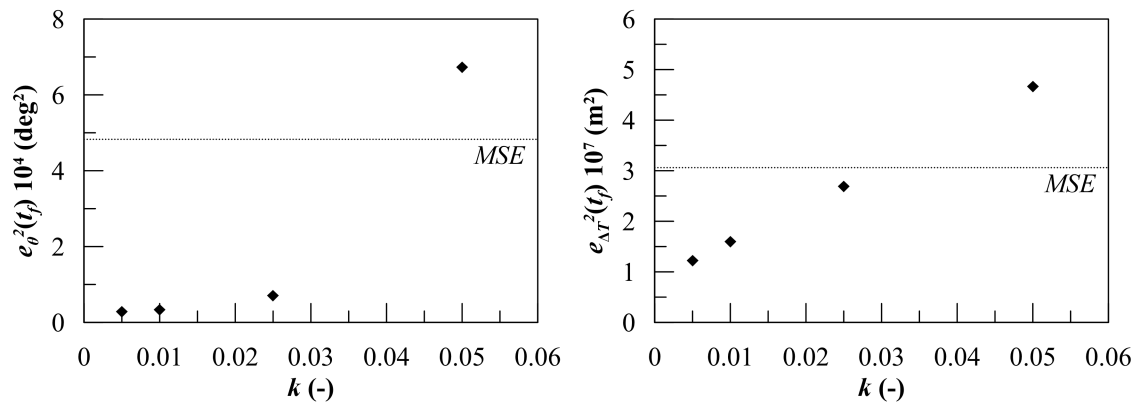


Figure 15. Trim angle and sinkage squared error at final stage in damage scenario B.

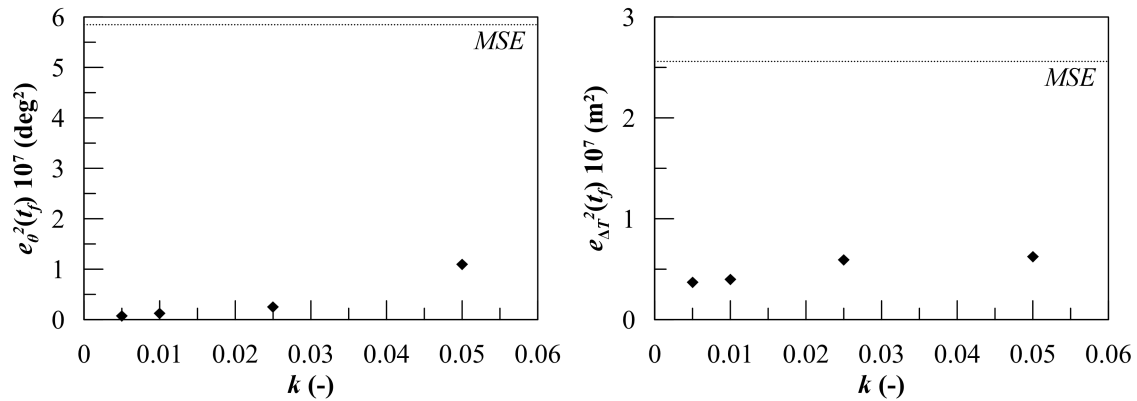


Figure 16. Trim angle and sinkage squared error at final stage in damage scenario C.

Increasing the value of k , the simulated data shows an increasing divergence from the results obtained assuming $k = 0.010$. The time step limitation algorithm allowed to take under control the late recognition of the new flooding rooms. Anyway, the adoption of higher time steps lead to an increasing delay in the intermediate stages of transient flooding. In detail, sinkage error ratios differ up to 0.04 and 0.08 in scenarios B and C, respectively, when $k = 0.05$ is applied. In scenario A, the heel angle overshoot is also overestimated up to 0.1 as k increases. Nevertheless, aiming to the onboard application, these errors may be still acceptable. In fact, they are lower or comparable to the ones coming from the flooding chains method that has been considered satisfactory [13].

The adoption of higher values of k is also encouraged by its positive repercussion on the computational effort, which is a key factor during a flooding emergency. Table 8 provides the whole computation time t_C corresponding to applied k values. Again the performances are related to a standard laptop. The simulation is 25–45 times faster than real-time evolution of progressive flooding for $k = 0.010$. Applying higher values, the simulation is 45–80 times faster than time-to-flood and more than 5 times faster than the LODE method (assuming a constant time step $dt = 0.5$ s) [14], providing comparable accuracy. Hence, although the DAE formulation lead to a moderate increase of system solution time on the single loop iteration, the application of the adaptive time step allows to reproduce the benchmark test case with a considerable computation time reduction compared with the previous LODE technique.

Table 8. Computation time required by the linearised Differential-Algebraic Equation (LDAE) method for the model-scale experiments.

Case	A				B				C			
k	0.005	0.010	0.025	0.050	0.005	0.010	0.025	0.050	0.005	0.010	0.025	0.050
t_F (s)	350.0	350.0	350.0	350.0	440.0	440.0	440.0	440.0	820.0	820.0	820.0	820.0
t_C (s)	21.1	13.3	7.4	5.3	48.3	30.8	11.2	9.7	28.9	18.5	10.8	10.2
$\%t_F$ (%)	6.0	3.8	2.1	1.5	11.1	7.0	2.5	2.2	3.5	2.3	1.3	1.2

5.2. Procedures Comparison

Adopting a k value of 0.010, it is than possible to compare on the benchmark case the proposed LDAE procedures with experimental results and the other simulation methods. In particular, Figures 17–19 provide a comparison of LDAE technique results with the ones coming from the experimental records, from the Pressure Correction (PC) technique [16] and from the methods (C1–C5) included in a benchmark study available in literature [26]. The results are in good agreement with PC technique as well as with experimental results on both the floating position parameters and the water levels inside the flooded rooms. Only the heel angle in scenario A has a different trend compared to the experimental one. However, such discrepancies are due to the quasi-static assumption that leads to an overestimation of the maximum heeling angle. Nevertheless, this behaviour is common to all the presented quasi-static integration techniques. For the analysed cases, the LDAE methodology reproduced with high fidelity the experimental progressive flooding pace, especially compared to C2, C4 and C5 methods.

Finally, it shall be noted that the benchmark test case does not present relevant issues connected to filled rooms. Means that the different size and vertical position of the connections lead to very different pressure losses in all the completely filled rooms. This is why the LODE technique was also found capable to provide sufficiently accurate results on the benchmark case [14].

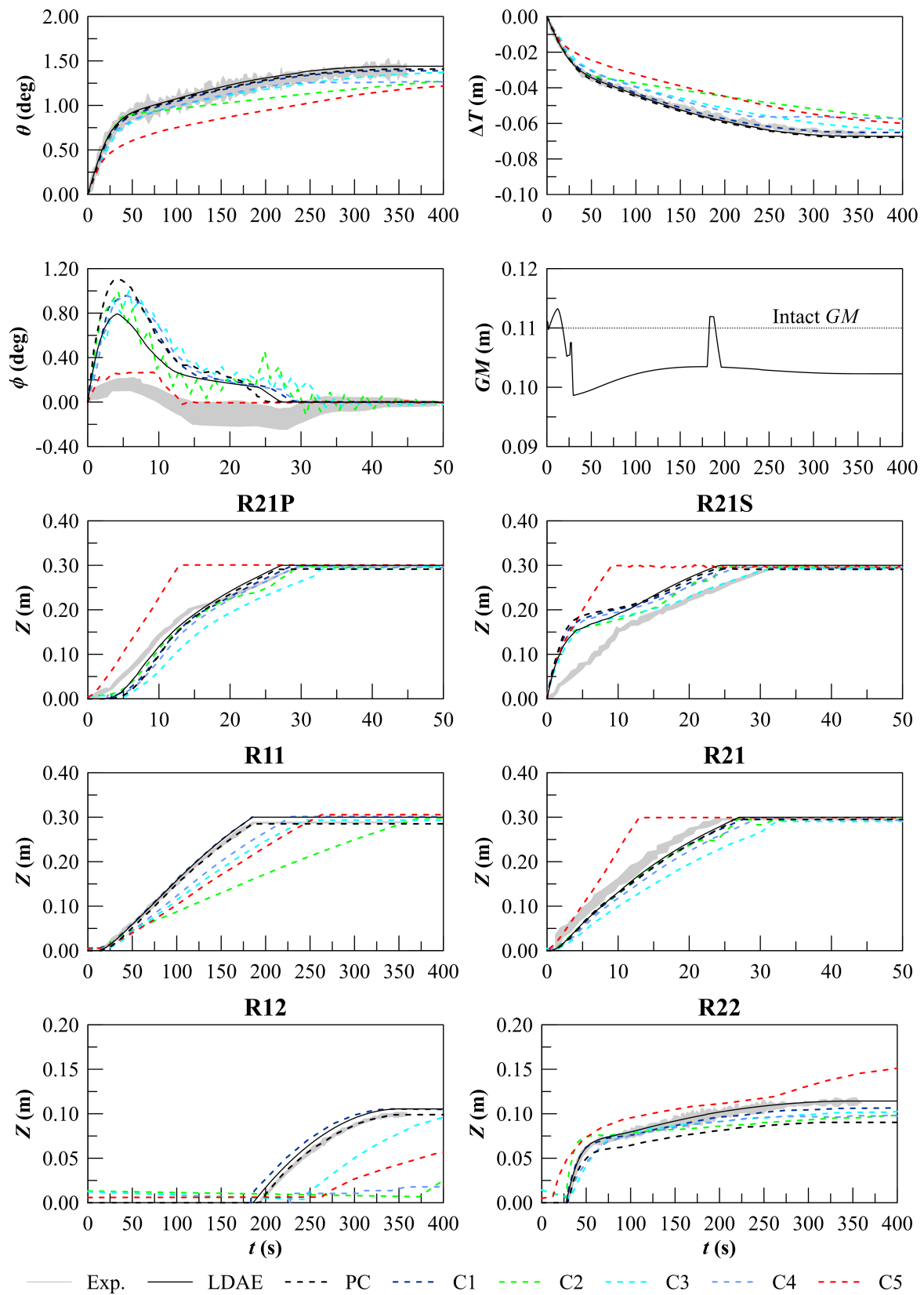


Figure 17. Comparison between the experimental record and the results coming from LDAE, Pressure Correction (PC) and C1–C5 techniques for case A.

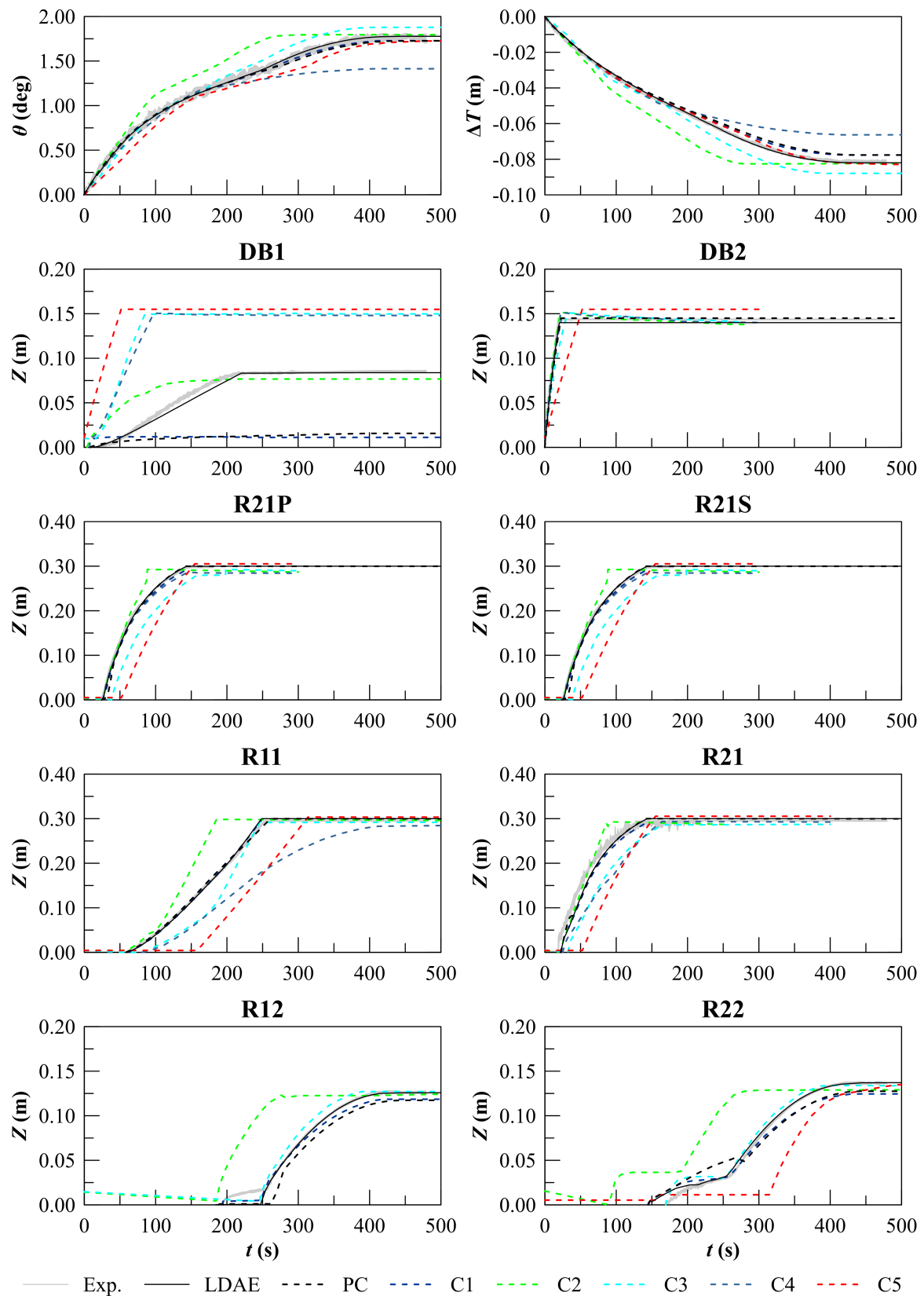


Figure 18. Comparison between the experimental record and the results coming from LDAE, PC and C1–C5 techniques for case B.

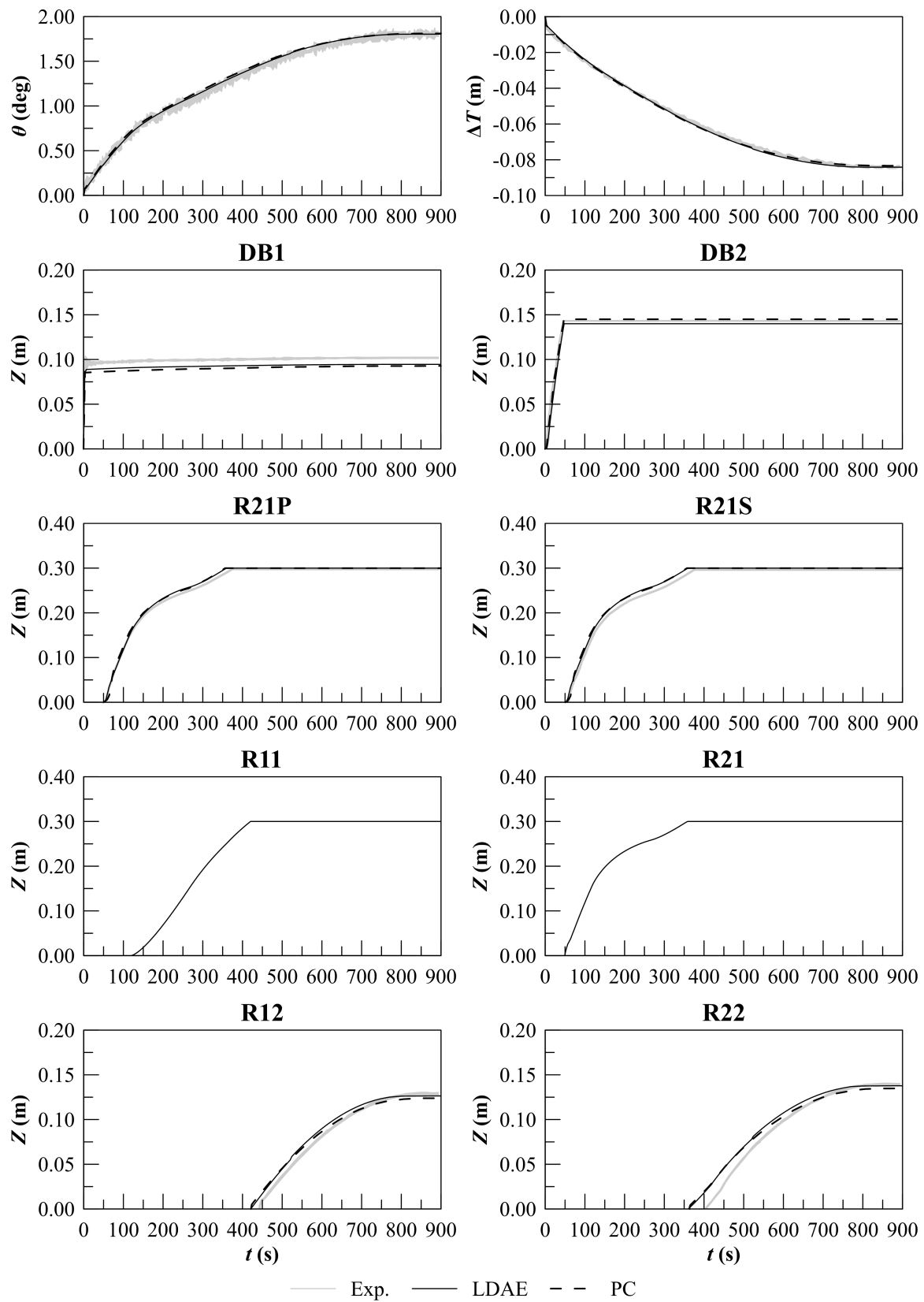


Figure 19. Comparison between the experimental record and the results coming from LDAE and PC techniques for case C.

6. Conclusions

The proposed linearised technique overcomes the restrictions, limiting the application of the previous LODE formulation in a real environment. In particular, the application of ODE formulation might lead to a relevant underestimation of the time-to-flood on long flooding chains. For decision support purposes, such an error caused by grouping procedure might be considered a conservative error. However, too short time-to-flood prediction might lead to rush emergency procedures, increasing the panic occurrence and affecting the vessel's evacuation. Moreover, in more complex subdivision layouts with multiple breaches, the erroneous pace related to an independent flooding chain caused by ODE formulation might affect the definition of the global flooding sequences, driving to the divergence between simulated and real events. Thus, these issues can strongly limit the LODE method application in a real environment. The adoption of the DAE system formulation allows more accurate and flexible simulation of progressive flooding scenarios including filled rooms and similar size connections. Moreover, the new method removes the need for grouping filled compartments leading to a very accurate prediction of the progressive flooding pace. Considering a single iteration of the main-loop, the new formulation does not imply a relevant drawback on computational effort: a moderate increase is caused by removing the grouping procedure that leads to a higher dimension of the equations system.

However, the computational effort of the overall simulation process has been over 5 times reduced through the application of an adaptive time step. The method based on the maximum value of the floodwater level derivatives provided stable and accurate simulation results and good computational performances, which could lead to an easy onboard application for decision support purposes. In fact, after the identification of the damage extension and position it could be possible to rapidly provide a simulation of the progressive flooding predicting the final ship fate. In addition, the proposed technique could be possibly adopted for generating large damage scenarios databases to be used during the design of a new ship. Moreover, the definition of the time step restriction procedure allowed to reach a satisfactory compromise between accuracy and computation time reduction on box-shaped geometries. This has been confirmed by the good agreement with experimental results and another more complex simulation technique present in literature.

Although the results here presented are promising, it is still required an application of the linearised simulation technique on a real large passenger ship. In fact, especially for slender hull forms, breaches located in the bow and stern region might call for additional limitation of the maximum time step to avoid the errors connected to the constant waterplane area assumption.

Author Contributions: Conceptualization, L.B. and F.M.; methodology, L.B.; software, L.B.; validation, L.B. and F.M.; formal analysis, F.M. and L.B.; investigation, L.B.; resources, L.B. and F.M.; data curation, L.B.; writing—original draft preparation, L.B.; writing—review and editing, F.M. and L.B.; visualization, F.M. and L.B.; supervision, F.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: There are no conflict of interest to disclose.

References

1. Dankowski, H.; Russel, P.; Krüger, S. New Insights Into the Flooding Sequence of the Costa Concordia Accident. In Proceedings of the 33rd International Conference on Ocean, Offshore and Arctic Engineering—OMAE 2014, San Francisco, CA, USA, 8–13 June 2014.
2. Nasso, C.; Bertagna, S.; Mauro, F.; Marinò, A.; Bucci, V. Simplified and advanced approaches for evacuation analysis of passenger ships in the early stage of design. *Brodogradnja* **2019**, *70*, 43–59. [[CrossRef](#)]
3. Santos, T.; Winkle, I.; Guedes Soares, C. Time domain modelling of the transient asymmetric flooding of Ro-Ro ships. *Ocean Eng.* **2002**, *29*, 667–688. [[CrossRef](#)]
4. Ruponen, P.; Pennanen, P.; Manderbacka, T. On the alternative approaches to stability analysis in decision support for damaged passenger ships. *WMU J. Marit. Aff.* **2019**, *18*, 477–494. [[CrossRef](#)]
5. Ruponen, P.; Pulkkinen, A.; Laaksonen, J. A method for breach assessment onboard a damaged passenger ship. *Appl. Ocean Res.* **2017**, *64*, 236–248. [[CrossRef](#)]

6. Trincas, G.; Braidotti, L.; De Francesco, L. Risk-Based System to Control Safety Level of Flooded Passenger Ship. *Brodogradnja* **2017**, *68*, 31–60. [CrossRef]
7. Acanfora, M.; Begovic, E.; De Luca, F. A Fast Simulation Method for Damaged Ship Dynamics. *J. Mar. Sci. Eng.* **2019**, *7*, 111. [CrossRef]
8. Ruponen, P.; Lindroth, D.; Pennanen, P. Prediction of survivability for decision support in ship flooding emergency. In Proceedings of the 12th International Conference on the Stability of Ships and Ocean Vehicles, Glasgow, UK, 14–19 June 2015.
9. Dankowski, H.; Krüger, S. A Fast, Direct Approach for the Simulation of Damage Scenarios in the Time Domain. In Proceedings of the 11th International Marine Design Conference—IMDC 2012, Glasgow, UK, 11–14 June 2012.
10. Lee, G. Dynamic orifice flow model and compartment models for flooding simulation of a damaged ship. *Ocean Eng.* **2015**, *109*, 635–653. [CrossRef]
11. Manderbacka, T.; Mikkola, T.; Ruponen, P.; Matusiak, J. Transient response of a ship to an abrupt flooding accounting for the momentum flux. *J. Fluids Struct.* **2015**, *57*, 108–126. [CrossRef]
12. Acanfora, M.; Cirillo, A. A simulation model for ship response in flooding scenario. *Proc. Inst. Mech. Eng. Part J. Eng. Marit. Environ.* **2017**, *231*, 153–164. [CrossRef]
13. Ruponen, P.; Larmela, M.; Pennanen, P. Flooding Prediction Onboard a Damage Ship. In Proceedings of the 11th International Conference on the Stability of Ships and Ocean Vehicles, Athens, Greece, 23–28 September 2012; pp. 391–400.
14. Braidotti, L.; Mauro, F. A New Calculation Technique for Onboard Progressive Flooding Simulation. *Ship Technol. Res.* **2019**, *66*, 150–162. [CrossRef]
15. ITTC. *Numerical Simulation of Capsize Behaviour of Damaged Ship in Irregular Seas*; Technical Report; 2017. Available online: <https://www.ittc.info/media/8149/75-02-07-044.pdf> (accessed on 21 May 2020).
16. Ruponen, P. Progressive Flooding of a Damaged Passenger Ship. Ph.D. Thesis, Helsinki University of Technology, Helsinki, Finland, 2007.
17. Ruponen, P.; Kurvinen, P.; Saisto, I.; Harras, J. Experimental and Numerical Study on Progressive Flooding in Full-Scale. *Int. J. Marit. Eng.* **2010**, *152*, 197–208. [CrossRef]
18. Rodrigues, J.; Lavrov, A.; Hinostroza, M.; Guedes Soares, C. Experimental and numerical investigation of the partial flooding of a barge model. *Ocean Eng.* **2018**, *169*, 586–603. [CrossRef]
19. Braidotti, L.; Marinò, A.; Bucci, V. On the Effect of Uncertainties on Onboard Progressive Flooding Simulation. In Proceedings of The 3rd International Conference on Nautical and Maritime Culture—CNM 2019, Naples, Italy, 14–15 November 2019; pp. 21–30. [CrossRef]
20. Braidotti, L.; Trincas, G.; Bucci, V. Analysis of the Influence of Pressure Field on Accuracy for Onboard Stability Codes. In Proceedings of The 19th International Conference on Ships and Maritime Research—NAV 2018, Trieste, Italy, 20–22 June 2018; pp. 80–87. [CrossRef]
21. Spouge, J. Technical Investigation of the Sinking of the Ro-Ro Ferry European Gateway. *Trans. RINA* **1986**, *128*, 49–72.
22. Rodrigues, J.; Guedes Soares, C. A generalized adaptive mesh pressure integration technique applied to progressive flooding of floating bodies in still water. *Ocean Eng.* **2015**, *110*, 140–151. [CrossRef]
23. Ruponen, P. Pressure-Correction Method for Simulation of Progressive Flooding and Internal Air flows. *Ship Technol. Res.* **2006**, *53*, 63–73. [CrossRef]
24. Ruponen, P. Adaptive time step in simulation of progressive flooding. *Ocean Eng.* **2014**, *78*, 35–44. [CrossRef]
25. Ruponen, P. *Model Test for the Progressive Flooding of a Box-Shaped Barge, Report M-292*; Technical Report; Helsinki University of Technology: Helsinki, Finland, 2006.
26. Van Walree, F.; Papanikolaou, A. Benchmark study of numerical codes for the prediction of time to flood of ships: Phase I. In Proceedings of the 9th International Workshop on Ship Stability, Hamburg, Germany, 30–31 August 2007.

